

Groupe de travail Réseau  
**Request for Comments : 2883**  
Catégorie : En cours de normalisation

Traduction Claude Brière de L'Isle

S. Floyd, ACIRI  
J. Mahdavi, Novell  
M. Mathis, Pittsburgh Supercomputing Center  
M. Podolsky, UC Berkeley  
juillet 2000

## Extension à l'option d'accusé de réception sélectif (SACK) pour TCP

### Statut du présent mémoire

La présente RFC spécifie un protocole de normalisation pour la communauté de l'Internet et appelle à des discussions et suggestions pour son amélioration. Prière de se reporter à l'édition en cours des "Internet Official Protocol Standards" (*normes officielles du protocole Internet*) (STD 1) pour connaître l'état de la normalisation et le statut du présent protocole. La distribution du présent mémoire n'est soumise à aucune restriction.

### Déclaration de Copyright

Copyright (C) The Internet Society (2000). Tous droits réservés.

### Résumé

La présente note définit une extension à l'option d'accusé de réception sélectif (SACK, *Selective Acknowledgment*) [RFC2018] pour TCP. La RFC 2018 spécifie l'utilisation de l'option SACK pour accuser réception des données hors séquence non couvertes par le champ TCP d'accusé de réception cumulatif. La présente note étend la RFC 2018 en spécifiant l'utilisation de l'option SACK pour accuser réception des paquets dupliqués. Elle suggère que lorsque des paquets dupliqués sont reçus, le premier bloc du champ d'option SACK peut être utilisé pour faire rapport des numéros de séquence du paquet qui a déclanché l'accusé de réception. Cette extension à l'option SACK permet à l'expéditeur TCP de déduire l'ordre des paquets reçus chez le receveur, ce qui permet à l'expéditeur d'en déduire quand il a retransmis inutilement un paquet. Un expéditeur TCP pourrait alors utiliser ces informations pour un fonctionnement plus robuste dans un environnement de paquets réarrangés [BPS99], de perte d'ACK, de duplication de paquet, et/ou de fin de temporisation précoce de retransmission.

## 1. Conventions et acronymes

Dans le présent document, les mots clés "DOIT", "NE DOIT PAS", "EXIGE", "DEVRA", "NE DEVRA PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDE", "PEUT", et "FACULTATIF" sont à interpréter comme décrit dans la [RFC2119].

## 2. Introduction

L'option d'accusé de réception sélectif (SACK) définie dans la RFC 2018 est utilisée par le receveur de données TCP pour accuser réception des blocs non contigus de données non couverts par le champ Accusé de réception cumulatif. Cependant, la RFC2018 ne spécifie pas l'utilisation de l'option SACK lorsque des segments dupliqués sont reçus. La présente note spécifie l'utilisation de l'option SACK lors de l'acquittement de la réception d'un paquet dupliqué [F99]. On utilise le terme D-SACK (pour SACK dupliqué) pour se référer à un bloc SACK qui fait rapport d'un segment dupliqué.

Le présent document ne fait aucun changement à l'utilisation par TCP du champ Accusé de réception cumulatif, ni à la décision du receveur TCP de \*quand\* envoyer un paquet d'accusé de réception. Le présent document ne concerne que le contenu de l'option SACK lorsque un accusé de réception est envoyé.

Cette extension est compatible avec les mises en œuvre courantes de l'option SACK dans TCP. C'est-à-dire que si un des nœuds d'extrémité TCP ne met pas en œuvre cette extension D-SACK et si l'autre nœud extrémité TCP la met en œuvre, on pense que cette utilisation de l'extension D-SACK par un des nœuds d'extrémité n'introduira pas de problème.

L'utilisation de D-SACK n'exige pas une négociation distincte entre un expéditeur et un receveur TCP qui ont déjà négocié la capacité SACK. L'absence d'une négociation séparée pour D-SACK signifie que le receveur TCP pourrait envoyer des blocs D-SACK lorsque l'expéditeur TCP ne comprend pas cette extension à SACK. Dans ce cas, l'expéditeur TCP va simplement éliminer tout bloc D-SACK, et traiter les autres blocs SACK dans le champ d'option SACK comme il le ferait normalement.

### 3. Le format d'option Sack défini dans la RFC2018

La RFC 2018 définit l'option SACK comme suit :

```

+-----+-----+
| Type=5 |Longueur|
+-----+-----+-----+-----+
|      Bord gauche du premier bloc |
+-----+-----+-----+-----+
|      Bord droit du premier bloc  |
+-----+-----+-----+-----+
|                                     |
/          . . .                      /
|                                     |
+-----+-----+-----+-----+
|      Bord gauche du bloc n        |
+-----+-----+-----+-----+
|      Bord droit du bloc n         |
+-----+-----+-----+-----+

```

L'option Accusé de réception sélectif (SACK) dans l'en-tête TCP contient un certain nombre de blocs SACK, où chaque bloc spécifie le bord droit et gauche d'un bloc de données reçu chez le receveur TCP. En particulier, un bloc représente un espace de séquence contiguë de données reçues et mises en file d'attente chez le receveur, où le "bord gauche" du bloc est le premier numéro de séquence du bloc, et le "bord droit" est le numéro de séquence qui suit immédiatement le dernier numéro de séquence du bloc.

La RFC2018 implique que le premier bloc SACK spécifie le segment qui a déclenché l'accusé de réception. D'après la RFC2018, lorsque le receveur des données choisit d'envoyer une option SACK, "le premier bloc SACK ... DOIT spécifier le bloc de données contiguës qui contient le segment qui a déclenché cet ACK, sauf si ce segment a avancé le champ Numéro d'accusé de réception dans l'en-tête".

Cependant, la RFC2018 ne traite pas de l'utilisation de l'option SACK lors de l'accusé de réception d'un segment dupliqué. Par exemple, la RFC2018 spécifie que "chaque bloc représente les octets de données reçus qui sont contigus et isolés". La RFC2018 spécifie de plus que "si elles sont envoyées, les options SACK DEVRAIENT être incluses dans tous les ACK qui n'accusent pas réception du plus fort numéro de séquence dans la file d'attente du receveur des données". La RFC2018 ne spécifie pas l'utilisation de l'option SACK lorsque un segment dupliqué est reçu, et le champ Accusé de réception cumulatif dans le ACK accuse réception de toutes les données dans la file d'attente du receveur des données.

### 4. Utilisation de l'option SACK pour faire rapport d'un segment dupliqué

Cette section spécifie l'utilisation des blocs SACK lorsque l'option SACK est utilisée pour faire rapport d'une duplication de segment. Lorsque D-SACK est utilisé, le premier bloc de l'option SACK devrait être un bloc D-SACK qui spécifie les numéros de séquence pour le segment dupliqué qui a déclenché l'accusé de réception. Si le segment dupliqué fait partie d'un plus grand bloc de données non contiguës dans la file d'attente de données du receveur, le bloc SACK suivant devrait alors être utilisé pour spécifier ce plus grand bloc. Des blocs SACK supplémentaires peuvent être utilisés pour spécifier des blocs de données non contigus supplémentaires, comme spécifié dans la RFC2018.

Les lignes directrices pour faire rapport des segments dupliqués sont résumées ci-dessous :

- (1) Un bloc D-SACK n'est utilisé que pour faire rapport d'une duplication d'une séquence de données contiguës reçues par le receveur dans le paquet le plus récent.
- (2) Chaque séquence de données contiguës dupliquée reçue fait l'objet d'un rapport dans au plus un bloc D-SACK. (C'est-à-dire que le receveur envoie deux blocs D-SACK identiques dans les paquets suivants seulement si le receveur reçoit deux segments dupliqués.)
- (3) Le bord gauche du bloc D-SACK spécifie le premier numéro de séquence de la séquence contiguë dupliquée, et le bord droit du bloc D-SACK spécifie le numéro de séquence qui suit immédiatement la dernière séquence dans la séquence contiguë dupliquée.
- (4) Si le bloc D-SACK rapporte une séquence contiguë dupliquée à partir d'un bloc (éventuellement plus grand) de données dans la file d'attente de données du receveur au dessus de l'accusé de réception cumulatif, alors le second bloc SACK dans cette option SACK devrait spécifier ce bloc de données (éventuellement plus grand).
- (5) À la suite des blocs SACK décrits ci-dessus pour le rapport des segments dupliqués, des blocs SACK supplémentaires peuvent être utilisés pour faire rapport de blocs de données supplémentaires, comme spécifié dans la RFC2018.

Noter que comme chaque segment dupliqué ne fait l'objet d'un rapport que dans un seul paquet ACK, les informations sur ce segment dupliqué seront perdues si ce paquet ACK est abandonnée dans le réseau.

#### 4.1 Rapport de segments complètement dupliqués

On illustre ces lignes directrices par trois exemples. Dans chaque exemple, on suppose que le receveur des données a d'abord reçu huit segments de 500 octets chacun, et a envoyé un accusé de réception avec le champ Accusé de réception cumulatif réglé à 4000 (en supposant que le premier numéro de séquence est zéro). Le bloc D-SACK est souligné dans chaque exemple.

##### 4.1.1 Exemple 1 : Rapport d'un segment dupliqué

Comme plusieurs paquets ACK sont perdus, l'expéditeur des données retransmet le paquet 3000-3499, et le receveur des données reçoit ensuite un segment dupliqué avec les numéros de séquence 3000-3499. Le receveur envoie un accusé de réception avec le champ Accusé de réception cumulatif réglé à 4000, et le premier, le bloc D-SACK qui spécifie les numéros de séquence 3000-3500.

Segment transmis	Segment reçu	ACK envoyé (blocs SACK inclus)
3000-3499	3000-3499	3500 (ACK abandonné)
3500-3999	3500-3999	4000 (ACK abandonné)
3000-3499	3000-3499	4000, SACK= <u>3000-3500</u>

##### 4.1.2 Exemple 2 : Rapport d'un segment déclassé et d'un segment dupliqué

Suite à un paquet de données perdu, le receveur reçoit un segment de données décalé, ce qui déclenche l'option SACK comme spécifié dans la RFC2018. À cause de plusieurs paquets ACK perdus, l'expéditeur retransmet alors un paquet de données. Le receveur reçoit le paquet dupliqué, et en fait rapport dans le premier bloc D-SACK :

Segment transmis	Segment reçu	ACK envoyé (blocs SACK inclus)
3000-3499	3000-3499	3500 (ACK abandonné)
3500-3999	3500-3999	4000 (ACK abandonné)
4000-4499	(paquet de données abandonné)	
4500-4999	4500-4999	4000, SACK=4500-5000 (ACK abandonné)
3000-3499	3000-3499	4000, SACK= <u>3000-3500, 4500-5000</u>

##### 4.1.3 Exemple 3 : Rapport de duplication d'un segment déclassé

À cause de la perte d'un paquet de données, le receveur reçoit deux segments décalés. Le receveur reçoit ensuite un segment dupliqué pour un de ces segments décalés :

Segment transmis	Segment reçu	ACK envoyé (blocs SACK inclus)
3500-3999	3500-3999	4000
4000-4499	(paquet de données abandonné)	
4500-4999	4500-4999	4000, SACK=4500-5000
5000-5499	5000-5499	4000, SACK=4500-5500
	(paquet dupliqué)	
	5000-5499	4000, SACK= <u>5000-5500, 4500-5500</u>

#### 4.2 Rapport de segments dupliqués partiels

Il se peut qu'un expéditeur transmette un paquet qui comporte un ou plusieurs sous-segments dupliqués – c'est-à-dire seulement une partie mais pas la totalité du paquet transmis est déjà arrivée chez le receveur. Cela peut se produire lorsque la taille des segments transmis par l'expéditeur augmente, par exemple lorsque la PMTU augmente au milieu d'une session TCP. Les lignes directrices de la Section 4 ci-dessus s'appliquent au rapport partiel aussi bien qu'à celui de segments dupliqués complets. Cette section donne des exemples de ces lignes directrices lors du rapport de segments dupliqués partiels

Lorsque l'option SACK est utilisée pour rapporter des segments dupliqués partiels, le premier bloc D-SACK fait rapport du

premier sous-segment dupliqué. Si le paquet de données dont il est accusé réception contient plusieurs sous-segments dupliqués partiels, seul le premier de ces sous-segment dupliqués est alors rapporté dans l'option SACK. On illustre cela par les exemples ci-dessous.

#### 4.2.1 Exemple 4 : Rapport d'un seule sous-segment dupliqué

L'expéditeur augmente la taille de paquet de 500 octets à 1000 octets. Le receveur reçoit ensuite un paquet de 1000 octets contenant un sous-segment de 500 octets qui a déjà été reçu et un qui ne l'était pas. Le receveur fait seulement rapport du sous-segment déjà reçu en utilisant un seul bloc D-SACK.

Segment transmis	Segment reçu	ACK envoyé (blocs SACK inclus)
500-999	500-999	1000
1000-1499	(retardé)	
1500-1999	(paquet de données abandonné)	
2000-2499	2000-2499	1000, SACK=2000-2500
1000-2000	1000-1499	1500, SACK=2000-2500
	1000-2000	2500, SACK= <u>1000-1500</u>

#### 4.2.2 Exemple 5 : Deux sous-segments dupliqués non contigus couverts par l'accusé de réception cumulatif

Après que l'expéditeur a augmenté la taille de son paquet de 500 à 1500 octets, le receveur reçoit un paquet contenant deux sous-segments non contigus dupliqués de 500 octets qui sont moins que le champ Accusé de réception cumulatif. Le receveur fait rapport du premier de ces segments dupliqués dans un seul bloc D-SACK.

Segment transmis	Segment reçu	ACK envoyé (blocs SACK inclus)
500-999	500-999	1000
1000-1499	(retardé)	
1500-1999	(paquet de données abandonné)	
2000-2499	(retardé)	
2500-2999	(paquet de données abandonné)	
3000-3499	3000-3499	1000, SACK=3000-3500
1000-2499	1000-1499	1500, SACK=3000-3500
	2000-2499	1500, SACK=2000-2500, 3000-3500
	1000-2499	2500, SACK= <u>1000-1500</u> , 3000-3500

#### 4.2.3 Exemple 6 : Deux sous-segments dupliqués non contigus non couverts par l'accusé de réception cumulatif

Cet exemple est similaire à l'exemple 5, sauf qu'après que l'expéditeur a augmenté la taille de paquet, le receveur reçoit un paquet contenant deux sous-segments non contigus dupliqués qui sont au dessus du champ Accusé de réception cumulatif, plutôt qu'en dessous. Le premier, un bloc D-SACK, fait rapport du premier sous-segment dupliqué, et le second, un bloc SACK, fait rapport du plus grand bloc de données non contiguës auquel il appartient.

Segment transmis	Segment reçu	ACK envoyé (blocs SACK inclus)
500-999	500-999	1000
1000-1499	(paquet de données abandonné)	
1500-1999	(retardé)	
2000-2499	(paquet de données abandonné)	
2500-2999	(retardé)	
3000-3499	(paquet de données abandonné)	
3500-3999	3500-3999	1000, SACK=3500-4000
1000-1499	(paquet de données abandonné)	
1500-2999	1500-1999	1000, SACK=1500-2000, 3500-4000
	2000-2999	1000, SACK=2000-2500, 1500-2000, 3500-4000
	1500-2999	1000, SACK= <u>1500-2000</u> , 1500-3000, 500-4000

### 4.3 Interaction entre D-SACK et PAWS

La [RFC1323] spécifie un algorithme de protection contre le retour à zéro des numéros de séquence (PAWS, *Protection Against Wrapped Sequence Numbers*). PAWS donne une méthode pour distinguer entre les numéros de séquence pour de nouvelles données, et les numéros de séquence provenant d'un cycle précédent à travers l'espace des numéros de séquence. Les segments dupliqués pourraient être détectés par PAWS comme appartenant à un cycle précédent à travers l'espace des

numéros de séquence.

La RFC1323 spécifie que pour de tels paquets, le receveur devrait envoyer un accusé de réception en réponse comme spécifié dans la RFC793, page 69, et abandonner le segment.

Comme PAWS exige encore d'envoyer un ACK, il n'y a pas d'interaction dommageable entre PAWS et l'utilisation de D-SACK. Le bloc D-SACK peut être inclus dans l'option SACK de l'ACK, comme mentionné à la Section 4, indépendamment de l'utilisation de PAWS par le receveur TCP, et indépendamment de la détermination par PAWS de la validité ou non du segment de données.

Les envoyeurs TCP qui reçoivent des blocs D-SACK devraient être conscients qu'un segment rapporté comme dupliqué pourrait éventuellement provenir d'un cycle antérieur à travers l'espace des numéros de séquence. Ceci est indépendant de l'utilisation de PAWS par le receveur des données TCP. On ne prévoit pas que cela puisse présenter de problèmes significatifs pour les envoyeurs qui utilisent les informations de D-SACK.

## 5. Détection de paquets dupliqués

Cette extension à l'option SACK permet au receveur de faire un rapport précis de la réception de données dupliquées. Comme chaque réception d'un paquet dupliqué fait l'objet d'un rapport sur un seul paquet ACK, la perte d'un seul ACK peut empêcher ces informations d'arriver jusqu'à l'envoyeur. De plus, on note que l'envoyeur peut ne pas nécessairement faire confiance au receveur pour lui envoyer des informations précises [SCWA99].

Afin que l'envoyeur vérifie que le premier bloc (D)SACK d'un accusé de réception acquitte en fait des données dupliquées, l'envoyeur devrait comparer l'espace des numéros de séquence dans le premier bloc SACK avec le ACK cumulatif qui est porté DANS LE MÊME PAQUET. Si l'espace de numéros de séquence du SACK est inférieur à celui de cet ACK cumulatif, cela indique que le segment identifié par le bloc SACK a été reçu plus d'une fois par le receveur. Une mise en œuvre NE DOIT PAS comparer l'espace de numéros de séquence dans le bloc SACK avec la variable d'état TCP `snd.una` (qui porte l'ACK cumulatif total) car il peut en résulter une mauvaise conclusion si les paquets ACK ont été réarrangés.

Si l'espace des numéros de séquence dans le premier bloc SACK est supérieur à celui de l'ACK cumulatif, l'envoyeur compare ensuite l'espace des numéros de séquence dans le premier bloc SACK avec l'espace des numéros de séquence dans le second bloc SACK, si il y en a un. Cette comparaison peut déterminer si le premier bloc SACK fait rapport de données dupliquées qui se tiennent au dessus de l'ACK cumulatif.

Les mises en œuvre de TCP qui suivent la [RFC2581] pourraient voir des paquets dupliqués dans chacune des quatre situations suivantes. Le présent document ne spécifie pas quelle action devrait entreprendre une mise en œuvre TCP dans ces cas. L'extension à l'option SACK permet simplement à l'envoyeur de détecter chacun de ces cas. Noter que ces quatre conditions ne constituent pas une liste exhaustive des cas possibles de paquets dupliqués, mais sont représentatives des cas les plus courants/vraisemblables. Des documents ultérieurs décriront les propositions expérimentales de réponses de l'envoyeur à la détection de retransmissions inutiles dues au réarrangement, à la perte des ACK, ou la fin de temporisation précoce de retransmissions.

### 5.1 Duplication par le réseau

Si un paquet est dupliqué dans le réseau, cette extension à l'option SACK peut l'identifier. Par exemple :

Segment transmis	Segment reçu	ACK envoyé (blocs SACK inclus)
500-999	500-999	1000
1000-1499	1000-1499 (dupliqué)	1500
	1000-1499	1500, SACK= <u>1000-1500</u>

Dans ce cas, le second paquet a été dupliqué dans le réseau. Un ACK contenant un bloc D-SACK qui est inférieur à son champ ACK et n'est pas identique à un segment retransmis précédemment indique une duplication par le réseau.

SANS D-SACK :

Si D-SACK n'a pas été utilisé et si le dernier ACK a été porté sur un paquet de données, l'envoyeur ne va pas savoir qu'un paquet a été dupliqué dans le réseau. Si D-SACK n'était pas utilisé et si aucun des deux derniers ACK n'était porté par un paquet de données, l'envoyeur pourra alors raisonnablement déduire que soit le paquet de données \*soit\* le paquet de ACK final a été dupliqué dans le réseau. La réception du paquet D-SACK donne à l'envoyeur une connaissance positive que ce paquet de données n'a pas été dupliqué dans le réseau (en supposant que le receveur ne ment pas).

Recherches en cours :

L'option SACK actuelle permet déjà à l'expéditeur d'identifier les ACK dupliqués qui n'acquittent pas de nouvelles données, mais l'option D-SACK donne à l'expéditeur une meilleure base pour déduire qu'un ACK dupliqué n'acquiesce pas de nouvelles données. Savoir qu'un ACK dupliqué n'acquiesce pas de nouvelles données permet à l'expéditeur de s'abstenir d'utiliser ces ACK dupliqués pour conclure à une perte de paquet (par exemple, une retransmission rapide) ou d'envoyer plus de données (par exemple, récupération rapide).

## 5.2 Fausse retransmission à cause de réarrangement

Si des paquets sont réarrangés dans le réseau d'une façon telle qu'un segment arrive avec un décalage de plus de trois paquets, l'algorithme de retransmission rapide de TCP va retransmettre le paquet déclassé. Un exemple de cela est montré ci-dessous :

Segment transmis	Segment reçu	ACK envoyé (blocs SACK inclus)
500-999	500-999	1000
1000-1499	(retardé)	
1500-1999	1500-1999	1000, SACK=1500-2000
2000-2499	2000-2499	1000, SACK=1500-2500
2500-2999	2500-2999	1000, SACK=1500-3000
1000-1499	1000-1499	3000
	1000-1499	3000, SACK= <u>1000-1500</u>

Dans ce cas, un ACK contenant un bloc SACK qui est inférieur à son champ ACK et identique à un segment retransmis précédemment indique un réarrangement significatif suivi par une fausse (inutile) retransmission.

SANS D-SACK :

Avec l'utilisation de D-SACK illustrée ci-dessus, l'expéditeur sait soit que la première transmission du segment 1000-1499 a été retardée dans le réseau, soit que la première transmission du segment 1000-1499 a été abandonnée et que la seconde transmission du segment 1000-1499 a été dupliquée. Comme aucun autre segment n'a été dupliqué dans le réseau, cette seconde option peut être considérée comme improbable.

Sans l'utilisation de D-SACK, l'expéditeur saurait seulement que soit la première transmission du segment 1000-1499 a été retardée dans le réseau, soit que un des segments de données ou l'ACK final a été dupliqué dans le réseau. Donc, l'utilisation de D-SACK permet à l'expéditeur de déduire de façon plus fiable que la première transmission du segment 1000-1499 n'a pas été abandonnée.

[AP99], [L99], et [LK00] notent que l'expéditeur pourrait détecter de façon non ambiguë une retransmission inutile avec l'utilisation de l'option d'horodatage. [LK00] propose un algorithme fondé sur l'horodatage qui minimise la sanction pour une retransmission inutile. [AP99] propose une heuristique pour détecter une retransmission inutile dans un environnement sans horodatage ni SACK. [L99] propose aussi un champ de deux bits comme solution de remplacement à l'option d'horodatage pour un marquage sans ambiguïté des trois premières retransmissions d'un paquet. Une idée similaire a été proposée dans [ISO8073].

Recherches en cours :

L'utilisation de D-SACK permet à l'expéditeur de détecter des cas (par exemple, lorsque aucun paquet d'ACK n'a été perdu) lorsque une retransmission rapide était due au réarrangement des paquets plutôt qu'à une perte de paquet. Cela permet à l'expéditeur TCP d'ajuster le seuil d'accusé de réception dupliqué, pour empêcher à l'avenir de telles retransmissions rapides inutiles. Couplé avec cela, lorsque l'expéditeur détermine, après coup, qu'il a fait une réduction de fenêtre inutile, il a l'option de "défaire" cette réduction dans la fenêtre d'encombrement en rétablissant ssthresh à la valeur de l'ancienne fenêtre d'encombrement, et de faire un redémarrage lent jusqu'à ce que la fenêtre d'encombrement ait atteint ce point.

Toute proposition de "défaire" une réduction de la fenêtre d'encombrement devrait traiter la possibilité que le receveur TCP puisse mentir dans ses rapports de paquets reçus [SCWA99].

## 5.3 Fin de temporisation de retransmission due à une perte d'ACK

Si une fenêtre entière de ACK est perdue, il va en résulter une fin de temporisation. On en donne un exemple ci-dessous :

Segment transmis	Segment reçu	ACK envoyé (blocs SACK inclus)
500-999	500-999	1000 (ACK abandonné)
1000-1499	1000-1499	1500 (ACK abandonné)
1500-1999	1500-1999	2000 (ACK abandonné)
2000-2499	2000-2499	2500 (ACK abandonné)
(fin de temporisation)		
500-999	500-999	2500, SACK= <u>500-1000</u>

Dans ce cas, tous les ACK sont abandonnés, d'où il résulte une fin de temporisation. Cette condition peut être identifiée parce que le premier ACK reçu après la fin de temporisation porte un bloc D-SACK qui indique que des données dupliquées ont été reçues.

#### SANS D-SACK :

Sans l'utilisation de D-SACK, l'expéditeur serait dans ce cas incapable de décider si des paquets de données ont été abandonnés.

#### Recherches en cours :

Pour un TCP qui met en œuvre une forme de contrôle d'encombrement de ACK [BPK97], cette capacité à distinguer des paquets de données abandonnés de paquets de ACK abandonnés serait particulièrement utile. Dans ce cas, la connexion pourrait mettre en œuvre un contrôle d'encombrement pour le chemin de retour (des ACK) indépendamment du contrôle d'encombrement sur le chemin direct (des données).

### 5.4 Temporisation de retransmission précoce

Si la temporisation de retransmission de l'expéditeur (RTO, *Retransmission TimeOut*) est trop courte, une temporisation de retransmission précoce peut survenir alors qu'aucun paquet n'a en fait été abandonné dans le réseau. En exemple en est donné ci-dessous :

Segment transmis	Segment reçu	ACK envoyé (blocs SACK inclus)
500-999	(retardé)	
1000-1499	(retardé)	
1500-1999	(retardé)	
2000-2499	(retardé)	
(fin de temporisation)		
500-999	(retardé)	
	500-999	1000
1000-1499	(retardé)	
	1000-1499	1500
	...	
	1500-1999	2000
	2000-2499	2500
	500-999	2500, SACK= <u>500-1000</u>
	1000-1499	2500, SACK= <u>1000-1500</u>
	...	

Dans ce cas, le premier paquet est retransmis à la suite de la fin de temporisation. Ensuite, la fenêtre originale de paquets arrive chez le receveur, résultant en des ACK pour ces segments. Ensuite, les retransmissions de ces segments arrivent, résultant en des ACK qui portent des blocs SACK qui identifient les segments dupliqués.

Cela peut être identifié comme une fin de temporisation de retransmission précoce parce que le ACK pour l'octet 1000 est reçu après la fin de temporisation sans informations de SACK, suivi par un ACK qui porte les informations de SACK (500-999) qui indiquent que le segment retransmis a déjà été reçu.

#### SANS D-SACK :

Si D-SACK n'était pas utilisé et si un des ACK dupliqués était porté sur un paquet de données, l'expéditeur ne saurait pas combien de paquets dupliqués ont été reçus. Si D-SACK n'était pas utilisé et si aucun des ACK dupliqués n'était porté sur un paquet de données, l'expéditeur aurait envoyé N paquets dupliqués, pour un certain N, et aurait reçu N ACK dupliqués. Dans ce cas, l'expéditeur pourrait raisonnablement déduire que certains paquets de données ou d'ACK ont été dupliqués dans le réseau, ou qu'une fin de temporisation de retransmission précoce s'est produite (ou que le receveur ment).

#### Recherches en cours :

Après que l'expéditeur a déterminé qu'une fin de temporisation de retransmission inutile (c'est-à-dire précoce) s'est

produite, il pourrait ajuster les paramètres de réglage du RTO, pour empêcher d'autres fins de temporisation de retransmission inutiles. Couplé avec cela, lorsque l'envoyeur détermine, après coup, qu'il a fait une réduction de fenêtre inutile, il a l'option de "défaire" cette réduction de la fenêtre d'encombrement.

## 6. Considérations pour la sécurité

Le présent document ne renforce ni n'affaiblit aucune des propriétés de sécurité actuelles de TCP.

## 7. Remerciements

Nous tenons à remercier Mark Handley, Reiner Ludwig, et Venkat Padmanabhan de leurs entretiens sur ces questions, et à manifester notre gratitude à Mark Allman pour ses retours utiles sur ce document.

## 8. Références

- [AP99] Mark Allman and Vern Paxson, "On Estimating End-to-End Network Path Properties", SIGCOMM 99, août 1999. URL " <http://www.acm.org/sigcomm/sigcomm99/papers/session7-3.html> ".
- [BPS99] J.C.R. Bennett, C. Partridge, and N. Shectman, "Packet Reordering is Not Pathological Network Behavior", IEEE/ACM Transactions on Networking, Vol. 7, n° du 6 décembre 1999, pp. 789-798.
- [BPK97] Hari Balakrishnan, Venkata Padmanabhan, and Randy H. Katz, "The Effects of Asymmetry on TCP Performance", Third ACM/IEEE Mobicom Conference, Budapest, Hungary, Sep 1997. URL : <http://www.cs.berkeley.edu/~padmanab/index.html#Publications>
- [F99] Floyd, S., "Re: TCP and out-of-order delivery", Message ID <199902030027.QAA06775@owl.ee.lbl.gov> to the end-to-end-interest mailing list, février 1999. URL : [http://www.aciri.org/floyd/notes/TCP\\_Feb99.email](http://www.aciri.org/floyd/notes/TCP_Feb99.email)
- [ISO8073] ISO/IEC, "Information-processing systems - Open Systems Interconnection - Connection Oriented Transport Protocol Specification", International Standard ISO/IEC 8073, décembre 1988.
- [L99] Reiner Ludwig, "A Case for Flow Adaptive Wireless links", Technical Report UCB//CSD-99-1053, mai 1999. URL : <http://iceberg.cs.berkeley.edu/papers/Ludwig-FlowAdaptive/>
- [LK00] Reiner Ludwig and Randy H. Katz, "The Eifel Algorithm: Making TCP Robust Against Spurious Retransmissions", SIGCOMM Computer Communication Review, V. 30, N. 1, janvier 2000. URL <http://www.acm.org/sigcomm/ccr/archive/ccr-toc/ccr-toc-2000.html>
- [RFC1323] V. Jacobson, R. Braden et D. Borman, "[Extensions TCP](#) pour de bonnes performances", mai 1992.
- [RFC2018] M. Mathis et autres, "Options d'[accusé de réception sélectif](#) sur TCP", octobre 1996. (*Remplace RFC1072 (P.S.)*)
- [RFC2581] M. Alman, V. Paxson et W. Stevens, "[Contrôle d'encombrement avec TCP](#)", avril 1999. (*Obsolète, voir RFC5681*)
- [SCWA99] Stefan Savage, Neal Cardwell, David Wetherall, Tom Anderson, "TCP Congestion Control with a Misbehaving Receiver", ACM Computer Communications Review, pp. 71-78, V. 29, n° du 5 octobre 1999. URL : <http://www.acm.org/sigcomm/ccr/archive/ccr-toc/ccr-toc-99.html>

## Adresse des auteurs

Sally Floyd  
AT&T Center for Internet Research at ICSI (ACIRI)  
téléphone : +1 510-666-6989  
mél : [floyd@aciri.org](mailto:floyd@aciri.org)  
URL : <http://www.aciri.org/floyd/>

Matthew Podolsky  
UC Berkeley Electrical Engineering & Computer Science Dept.  
téléphone : 510-649-8914  
mél : [podolsky@eecs.berkeley.edu](mailto:podolsky@eecs.berkeley.edu)  
URL : <http://www.eecs.berkeley.edu/~podolsky>

Jamshid Mahdavi  
Novell  
téléphone : 1-408-967-3806  
mél : [mahdavi@novell.com](mailto:mahdavi@novell.com)

Matt Mathis  
Pittsburgh Supercomputing Center  
téléphone : 412 268-3319  
mél : [mathis@psc.edu](mailto:mathis@psc.edu)  
URL : <http://www.psc.edu/~mathis/>

## **Déclaration de droits de reproduction**

Copyright (C) The Internet Society (2000). Tous droits réservés.

Le présent document et ses traductions peuvent être copiés et fournis aux tiers, et les travaux dérivés qui les commentent ou les expliquent ou aident à leur mise en œuvre peuvent être préparés, copiés, publiés et distribués, en tout ou partie, sans restriction d'aucune sorte, pourvu que la déclaration de droits de reproduction ci-dessus et le présent paragraphe soient inclus dans toutes les copies et travaux dérivés. Cependant, le présent document lui-même ne peut être modifié d'aucune façon, en particulier en retirant la notice de droits de reproduction ou les références à la Internet Society ou aux autres organisations Internet, excepté autant qu'il est nécessaire pour les besoins du développement des normes Internet, auquel cas les procédures de droits de reproduction définies dans les procédures des normes Internet doivent être suivies, ou pour les besoins de la traduction dans d'autres langues que l'anglais.

Les permissions limitées accordées ci-dessus sont perpétuelles et ne seront pas révoquées par la Internet Society ou ses successeurs ou ayant droits.

Le présent document et les informations y contenues sont fournies sur une base "EN L'ÉTAT" et le contributeur, l'organisation qu'il ou elle représente ou qui le/la finance (s'il en est), la INTERNET SOCIETY et la INTERNET ENGINEERING TASK FORCE déclinent toutes garanties, exprimées ou implicites, y compris mais non limitées à toute garantie que l'utilisation des informations ci-encloses ne violent aucun droit ou aucune garantie implicite de commercialisation ou d'aptitude à un objet particulier.

## **Remerciement**

Le financement de la fonction d'édition des RFC est actuellement fourni par l'Internet Society.